TECHNICAL DOCUMETNATION

PEEK EWA team 4

Team members:
Lisa Welling
Yosef van der Kleij
Dylan Worseling
Samy Farahat
Carolijn Pieterse

Table of context

1. Introduction

In 2023 Marius Peek gave us the assignment to make a web-application which should connect experts in their work field to existing companies who seek out Mr. Peeks help to find an expert who is interested and well qualified for the position this company needs for their project.

Marius Peek had the idea for this website because it was too difficult to keep track of which companies seek an expert to work on a project and who would fit the description for this project.

This web-application is a tool for Mr. Peek so he see which experts match with certain companies, and he can then reach out to both parties and see if they fit well together.

Our web-application is about finding those matches between a company and an expert. An admin, in this case Marius Peek, can see these matches and email both parties. This happens outside of the website.

Experts and companies don’t get to choose to whom their being matched with. There is an algorithm which decides that an expert and a company are a match.

2. Product vision

The most important feature of the web-application is the matching algorithm which matches experts with projects. As an expert or company you cannot see to whom you’re matched with. Only an admin can see the matches. We decided that an admin can only see the top 10 matches. With this matching system we calculate how many aspects the company and expert have in common, so that the company and expert fit well together.

An admin can see all the experts and all the projects of different companies. If you´re an expert you can only see the other experts and projects, but other than that there is no further action to be taken from the expert side after they have been registered.

An admin has to register the expert. The admin makes an empty account for the expert, and later the expert can add all their information in their profile.

For an admin its very important that they can add new projects onto the website. There is a feature only visible for admins that does exactly that. The admin has to fill out some information about the company and the project. And then creates the project.

The target user of this website are the experts. People whom have lots of experience in their line of work, and seek a project where they can use these skills, and where the company fits with the expert.

Alternatives for this website are:

- Indeed
- LinkedIn

But these websites don’t match projects or companies to experts based on how well the expert fits in that work environment, and in our website we do account for that.

The feature from this website that we could see growing in the future is the amount of people that would register. In the current state of the website it is only possible for an expert to sign up, by joining one of Mr. Peek his events where you can then get an invitation link for the website. If the website were to grow in the future, and Marius wants more experts to join the website, there needs to be an alternative for the invitation link system. Because people can not sign up by themselves, the amount of experts on the website stay rather low. The same goes for projects. Companies with projects can only sign up via Mr. Peek. They give their information to Mr. Peek and then he puts the project up on the website. But if the website were to grow, companies should be able to put their projects up on the website themselves.

The downside to this is that it would be much harder to control which people are actually experts, and have lots of experience. It would be much harder to keep out the people who don’t have that much experience, but sign up anyways.

3. Summary epic user stories

1. User story: Matching algorithm

Git: https://gitlab.fdmci.hva.nl/se-ewa/2023-2024-1/peek-4/-/issues/31

Accept criteria:

- 5 matched projects for each expert
- Matching on occupation, what position does the project need, years of experience etc (check requirements)

This user story focusses on the matching algorithm which matches experts and projects. We decided to have a maximum of 5 projects per expert, otherwise an expert could potentially match with hundreds of projects. Now an expert only matches with the 5 projects that fit best.

We match on the requirements given by Mr. Peek. Such as years of experience, knowledge and occupation.

We also made a different user story which focusses more on the details of this user story instead of the global lines of the matching algorithm.

2. User story: Make project roles more specific for matching algorithm accuracy

Git: https://gitlab.fdmci.hva.nl/se-ewa/2023-2024-1/peek-4/-/issues/39

Accept criteria:

- Be able to utilize existing input autocomplete components, for the sector tools and years of experience.
- Assign weights to categories for customization of matching requirements.

This user story belongs to the Matching algorithm user story. We received feedback from Mr. Peek and we came to the conclusion that we needed to implement weights for each item to be matched on.

4. Navigable Class Diagram

4.1 Full backend navigable class diagram

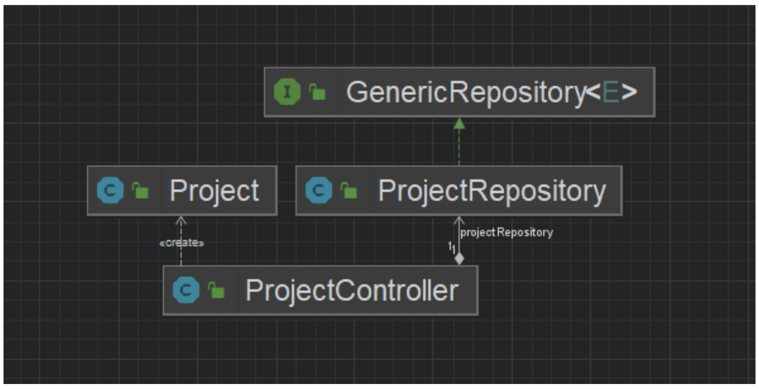
https://gitlab.fdmci.hva.nl/se-ewa/2023-2024-1/peek-4/-/blob/test/Technical%20Documentation/navigableClassDiagram.png?ref_type=heads

4.1 Domain entities navigable class diagram

https://gitlab.fdmci.hva.nl/se-ewa/2023-2024-1/peek-4/-/blob/test/Technical%20Documentation/domainClassDiagram.drawio.png?ref_type=heads

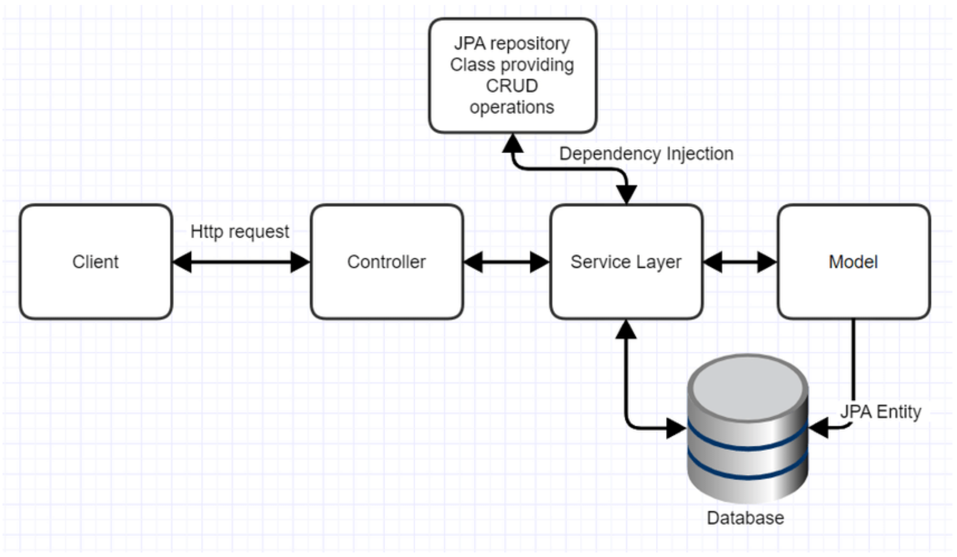
4.2 Structure and dependencies

This an example of the structure and dependencies in the backend for all the entities with model, repository and controller. In this example we used Project.



4.3 Use of external interface

This is the general flow for all entities in the backend with use of a external interface (Database).



5. Design Theme

5.1 Hashing

We wanted to safely store passwords of all the users in the database. We decided we opted for hashing, because this ensures that not just anybody can get your password. We are using hashing algorithm SHA-256. We also could have used MD5, but since this algorithm is the most known, we thought it would also be the most insecure algorithm.

5.2 REST API

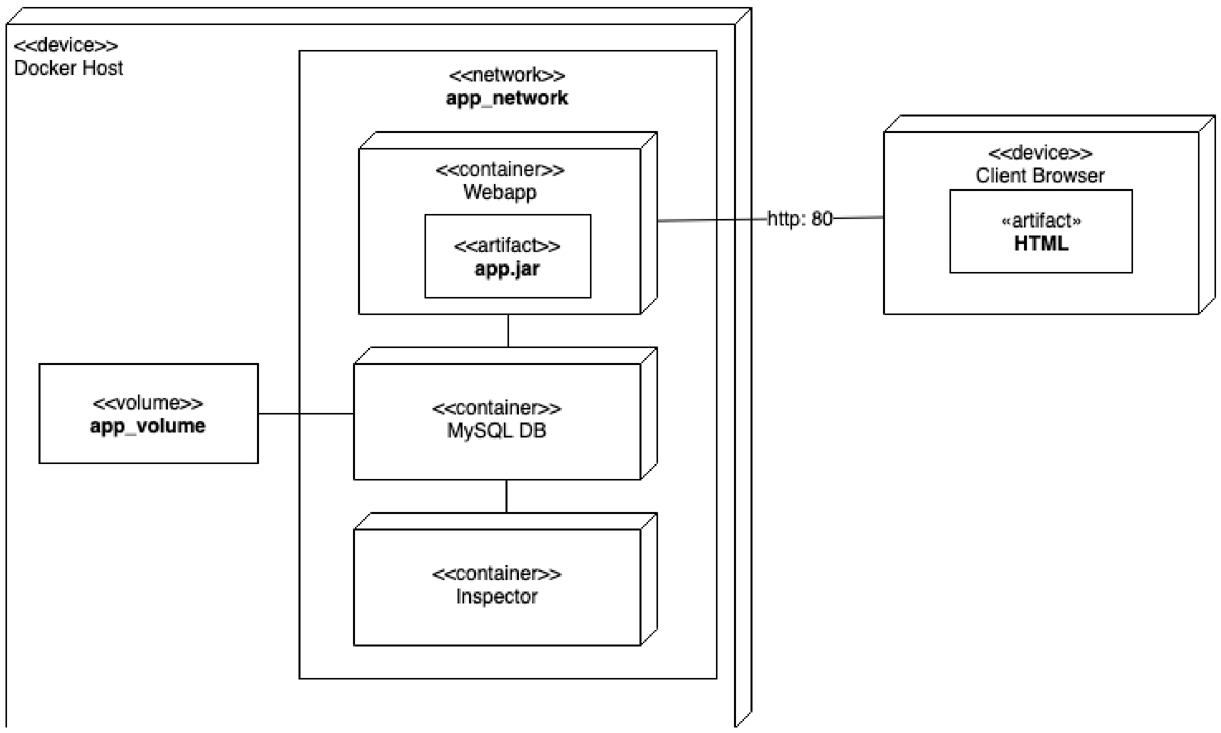
For the backend we are using SpringBoot with a MySQL database. For some queries we used namedQueries, that are placed in a model (e.g. User). But mostly we made the queries in the repository, We decided to do this because then the variables and the query itself were both in the same method. Whereas with namedQueries the variables and the query were in two different classes, which makes reading what the query is supposed to get or post more unclear.

5.3 External interfaces

It was already mentioned that we are using a MySQL database. We use SpringBoot for making tables, relations, primary keys, foreign keys, auto generated values etc.

We chose not to work with an in-memory database such as H2, because we figured that if the website became popular, then the MySQL database could easily handle a larger dataset. Although H2 is much faster than MySQL, we decided to work with MySQL, also because we had worked with SQL in the past.

6. Deployment Diagram



We use Docker and Render to deploy our website. We have an SQL database which communicates with the backend of our application.

7. Analytical Reflection

We decided to make a web application that has a simple design so that it is easy to see where to click and what to do. We also decided to focus on matching, because that was the main objective of Mr Peek.

7.1 Further improvements

The website could be improved by adding a chat function between a company and an expert. After the matching is done, there is no further action to be taken on the website. But Mr Peek still has to email both sides, to make sure the match is good, and that the experts is being signed on for the project. This could also be implemented in the website.

We asked Mr Peek what Experts could do on their page. They should only be able to view projects and to see other experts. This is what we implemented, and we also added a like function for the expert, so that they can like a project they might be interested in.